# One-sided Frank-Wolfe algorithms for saddle problems

**Vladimir Kolmogorov** [1] **Thomas Pock** [2]

## Abstract

We study a class of convex-concave saddle-point problems of the form $\min_x \max_y \langle Kx, y\rangle + f_{\mathcal{P}}(x) - h^*(y)$ where $K$ is a linear operator, $f_{\mathcal{P}}$ is the sum of a convex function $f$ with a Lipschitz-continuous gradient and the indicator function of a bounded convex polytope $\mathcal{P}$, and $h^*$ is a convex (possibly nonsmooth) function. Such problem arises, for example, as a Lagrangian relaxation of various discrete optimization problems. Our main assumptions are the existence of an efficient *linear minimization oracle* (`lmo`) for $f_{\mathcal{P}}$ and an efficient *proximal map* (`prox`) for $h^*$ which motivate the solution via a blend of proximal primal-dual algorithms and Frank-Wolfe algorithms. In case $h^*$ is the indicator function of a linear constraint and function $f$ is quadratic, we show a $O(1/n^2)$ convergence rate on the dual objective, requiring $O(n \log n)$ calls of `lmo`. If the problem comes from the constrained optimization problem $\min_{x\in\mathbb{R}^d}\{f_{\mathcal{P}}(x) \mid Ax - b = 0\}$ then we additionally get bound $O(1/n^2)$ both on the primal gap and on the infeasibility gap. In the most general case, we show a $O(1/n)$ convergence rate of the primal-dual gap again requiring $O(n \log n)$ calls of `lmo`. To the best of our knowledge, this improves on the known convergence rates for the considered class of saddle-point problems. We show applications to labeling problems frequently appearing in machine learning and computer vision.

## 1. Introduction

In this paper, we consider the following class of saddle-point problems:

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \mathcal{L}(x, y) := \langle Kx,\, y\rangle + f_{\mathcal{P}}(x) - h^*(y) \qquad (1)$$

where $\mathcal{X}, \mathcal{Y}$ are finite dimensional spaces, equipped with an inner product $\langle\cdot,\,\cdot\rangle$ and $K : \mathcal{X} \to \mathcal{Y}$ is a bounded linear operator with operator norm $L_K = \|K\|$. Usually the underlying spaces are the standard Euclidean spaces $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathbb{R}^m$.

The functions $f_{\mathcal{P}}(x)$ and $h^*$ are convex, lower semicontinuous functions.

For a differentiable convex function $f$ we say that it has a Lipschitz continuous gradient if there exists a constant $L_f \geq 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| \leq L_f \|x - y\|, \quad \forall x, y \in \mathcal{X}.$$

Moreover, the function $f$ is called strongly convex with strong convexity parameter $\mu_f > 0$ if $f(y) \geq f(x) + \langle \nabla f(x),\, y - x\rangle + \frac{\mu_f}{2}\|y - x\|^2$ for all $x, y \in \mathcal{X}$.

We make the following important structural assumptions on the functions $f_{\mathcal{P}}(x)$ and $h^*$:

- The function $f_{\mathcal{P}}(x)$ has the following composite form

  $$f_{\mathcal{P}}(x) = f(x) + \delta_{\mathcal{P}}(x),$$

  where $f$ is a convex function with a $L_f$-Lipschitz continuous gradient and $\delta_{\mathcal{P}}$ is the indicator function of a convex polytope $\mathcal{P} \subset \mathcal{X}$. For this polytope, we assume the existence of an efficient **linear minimization oracle** (`lmo`), which means that for any $a \in \mathcal{X}^*$, one can efficiently solve

  $$\texttt{lmo}_{\mathcal{P}}(a) \in \arg\min_{x\in\mathcal{P}} \langle a,\, x\rangle.$$

  This is for example the case if $\mathcal{P}$ is the polytope arising from LP relaxations of MAP-MRF problems in a tree-structured graph, where the above problem can be solved efficiently using dynamic programming.

- The function $h^*$ is a convex function which allows to efficiently compute its **proximal map** (`prox`), which for any $\bar{y} \in \mathcal{Y}$ and $\tau > 0$ is defined as

  $$\texttt{prox}_{\tau h^*}(\bar{y}) = \arg\min_{y\in\mathcal{Y}} \frac{1}{2\tau}\|y - \bar{y}\|^2 + h^*(y).$$

  Important examples of $h^*$ which allow for an efficient proximal map include quadratic functions and various norms. If $h^* = \delta_C$ i.e. the indicator functions of some convex set $C$ the proximal map reduces the orthogonal projection operator.

[1] Institute of Science and Technology Austria [2] Institute of Computer Graphics and Vision, Graz University of Technology. Correspondence to: Vladimir Kolmogorov <vnk@ist.ac.at>, Thomas Pock <pock@icg.tugraz.at>.

*Proceedings of the 38th International Conference on Machine Learning*, PMLR 139, 2021. Copyright 2021 by the author(s).

An important special case of problem (1) is given by

$$\min_{x\in\mathbb{R}^d}\max_{y\in\mathbb{R}^d}\mathcal{L}(x,y) := f_{\mathcal{P}}(x) + \langle y,\, Ax - b\rangle \qquad (2)$$

where $A$ is a matrix and $b$ is a vector of appropriate dimensions. This corresponds to the problem of minimizing $f_{\mathcal{P}}(x)$ subject to the linear constraint $Ax - b = 0$.

**Primal and dual problems** We denote by $(x^\star, y^\star)$ a saddle point of problem (1), which satisfies

$$\mathcal{L}(x^\star, y) \le \mathcal{L}(x^\star, y^\star) \le \mathcal{L}(x, y^\star), \quad \forall (x,y)\in\mathcal{X}\times\mathcal{Y}.$$

Throughout the paper we denote the primal and dual problems respectively by

$$F(x) = \max_{y\in\mathcal{Y}}\mathcal{L}(x,y), \quad H(y) = \min_{x\in\mathcal{X}}\mathcal{L}(x,y).$$

We assume that strong duality holds, that is

$$H(y^\star) = \max_{y\in\mathcal{Y}} H(y) = \max_{y\in\mathcal{Y}}\min_{x\in\mathcal{X}}\mathcal{L}(x,y) = \min_{x\in\mathcal{X}} F(x) = F(x^\star)$$

Some of the results will also assume coercivity of a function. We recall that a proper function $\phi(z)$ is called coercive if $\lim_{||z||\to\infty}\phi(z) = \infty$.

**Contributions** The algorithms we propose here are based on inexact proximal algorithms, which allow for an approximate evaluation of the proximal maps. For this we make use of efficient variants of the Frank-Wolfe algorithm that offer a linear convergence rate on the proximal subproblems. In summary, after $O(n\log n)$ calls to lmo we achieve the following guarantees.

- If function $f$ is linear or quadratic, $h^*$ is the indicator function of a linear constraint, and function $-H(y)$ is coercive then we obtain accuracy $O(1/n^2)$ on the dual objective $H$. If in addition the problem has the form of eq. (2) then we obtain bound $O(1/n^2)$ both on the primal gap $f_{\mathcal{P}}(x) - f_{\mathcal{P}}(x^\star)$ and on the infeasibility gap $||Ax - b||$.
- In the most general case, we obtain accuracy $O(1/n)$ on the dual objective (and also on the primal objective assuming that $\operatorname{dom} h^* \subseteq \mathcal{Y}$ is compact).

To the best of our knowledge these rates improve on the so far known rates for the class of saddle-point problems considered in this paper. In particular, for the problem in eq. (2) previous works (described later in Sec. 1.2) after $n$ calls to lmo obtained accuracy $O(1/n)$ on the dual objective and bound $O(1/\sqrt{n})$ on both $f_{\mathcal{P}}(x) - f_{\mathcal{P}}(x^\star)$ and $||Ax - b||$.

### 1.1. Motivating example

An important application, which also serves as the main motivation for the class of saddle-point problems studied in this paper, is given by the Lagrangian relaxation of discrete optimization problems. To form such relaxation, one needs to first encode discrete variables via Boolean indicator variables $X \in \{0,1\}^d$, and then express a difficult optimization problem as a sum of tractable subproblems:

$$\min_{X\in\{0,1\}^d}\ \sum_{t\in T} f_t(X_{A_t}) \qquad (3)$$

Here $T$ is the set of terms where each term $t$ is specified by a subset of variables $A_t \subseteq [d]$ and a function $f_t : \{0,1\}^{A_t} \to \mathbb{R}\cup\{+\infty\}$ of $|A_t|$ variables. Vector $X_{A_t}$ is the restriction of vector $X \in \mathbb{R}^d$ to $A_t$. The arity $|A_t|$ of function $f_t$ can be arbitrarily large, however we assume the existence of an efficient *min-oracle* that for a given vector $Y \in \mathbb{R}^{A_t}$ computes $X \in \underset{X\in\{0,1\}^{A_t}}{\arg\min}\ [f_t(X) + \langle X, Y\rangle]$ together with the cost $f_t(X)$. For example, this holds if $f_t(\cdot)$ corresponds to a MAP-MRF inference problem in a tree-structured graph.

Next, we can form a Lagrangian relaxation of the problem by treating $X$ and $X_{A_t}$ for $t\in T$ as independent variables and introducing Lagrange multipliers $Y_v^t$ for constraints of the form $X_v = (X_{A_t})_v$. This relaxation can be easily formulated as a saddle problem (1) where function $f(x)$ is linear and $h^*(y)$ is the indicator function of a linear constraint on $y$; we refer to (Swoboda & Kolmogorov, 2019) or the suppl. material for details.

As an example, the MAP-MRF inference problem on an undirected graph can be cast in the framework above by decomposing the graph into tree-structured subproblems. It is well-known that the Lagrangian relaxation is equivalent to a standard LP relaxation, aka the local polytope relaxation (Komodakis et al., 2011; Savchynskyy, 2019). MAP-MRF problems find numerous applications in machine learning and computer vision (Blake et al., 2011). In more recent work, they also appears as the final inference layer in deep convolutional neural networks (Knöbelreiter et al., 2020).

### 1.2. Related work

Saddle point problems in the form of (1) can be solved by a large number of proximal primal-dual algorithms (see for example the recent work (Condat et al., 2019) for a very comprehensive overview) as soon as the proximal maps for both the primal and dual functions can be solved efficiently. On the other hand, Gidel et al. (2017) proposed an extension of the Frank-Wolfe algorithm to saddle-point problems $\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}}\mathcal{L}(x,y)$ by assuming the existence of an efficient linear minimization oracle for the product space $\mathcal{X}\times\mathcal{Y}$ (which is assumed to be a compact set). In this paper, we are assuming the existence of an efficient linear minimization oracle just on the primal and an efficient proximal map on the dual. Therefore, our algorithms somewhat stand between the two aforementioned techniques.

Our algorithms rely on the inexact accelerated proximal gradient algorithm of Aujol & Dossal (2015) and the inexact primal-dual algorithm of Rasch & Chambolle (2020). Note that the first method only generates a dual sequence $\{y_n\}$. We extend the method and the analysis to also generate a primal sequence $\{x_n\}$, which is needed to solve the saddle problem (1).

Several authors studied a special case of (1) given in (2), or equivalently the problem of minimizing function $f_\mathcal{P}(x) = f(x) + \delta_\mathcal{P}(x)$ subject to linear constraints $Ax = b$ (Gidel et al., 2018; Liu et al., 2019; Yurtsever et al., 2018). (The last paper actually considered a more general class of saddle problems). Papers (Liu et al., 2019; Yurtsever et al., 2018) achieve an accuracy of $O(n^{-1/2})$ after $n$ iterations on the primal and infeasibility gaps, where (Yurtsever et al., 2018) uses one `lmo` call per iteration and (Liu et al., 2019) uses $O(k^2)$ `lmo` calls at $k$-th iteration assuming that a standard Frank-Wolfe method is employed. Note, the papers above do not give bounds on suboptimality gaps of the dual function $H$; instead, (Gidel et al., 2018; Liu et al., 2019) bound residuals of the *augmented* Lagrangian, which is not directly related to the residuals of the Lagrangian in eq. (2). A similar but slightly more general class of composite optimization problems was also recently considered in (Silveti-Falls et al., 2020). In a setting similar to our paper ($\nabla f$ is Lipschitz continuous) they show $O(n^{-1/3})$ accuracy on Lagrangian values after $n$ calls to `lmo`.

Frank-Wolfe algorithms for saddle-point problems have also been used in (Argyriou et al., 2014; Lan & Zhou, 2016). The former paper achieved a $O(n^{-1/2})$ convergence rate on a rather general class of constrained optimization problems. The paper (Lan & Zhou, 2016) shares some high-level similarities with our approach (such as solving smoothed primal subproblem to a given accuracy with Frank-Wolfe), but uses a different smoothing strategy that requires both primal and dual domains to be compact. This assumption rules out many interesting applications, including the one considered in Section 1.1.

There is a large body of literature on the special case of problem (1) corresponding to Lagrangian relaxation of discrete optimization problems, see e.g. (Storvik & Dahl, 2000; Schlesinger & Giginyak, 2007; Johnson et al., 2007; Ravikumar et al., 2010; Jojic et al., 2010; Savchynskyy et al., 2011; Schmidt et al., 2011; Komodakis et al., 2011; Martins et al., 2011; Savchynskyy et al., 2012; Luong et al., 2012; Schwing et al., 2012; 2014; Swoboda & Kolmogorov, 2019). Some of these methods apply only to MAP inference problems in pairwise (or low-order) graphical models, because they need to compute marginals in tree-structured subproblems (Johnson et al., 2007; Jojic et al., 2010; Savchynskyy et al., 2012) or because they explicitly exploit the fact that the relaxation can be described by polynomial many constraints (Schmidt et al., 2011; Martins et al., 2011; Schwing et al., 2012; 2014). The papers (Jojic et al., 2010; Savchynskyy et al., 2011) obtained accuracy $O(1/n)$ on the dual objective after $n$ iterations, by applying accelerated gradient methods (Nesterov, 1983).

The first method that we develop can be viewed as an extension of the technique in (Swoboda & Kolmogorov, 2019), which applied an inexact proximal point algorithm (PPA) to the dual objective. In contrast to (Swoboda & Kolmogorov, 2019), we apply an accelerated version of inexact PPA, specify to which accuracy the subproblems need to be solved, and analyze the convergence rate.

### 1.3. Notation for approximate solutions and organization of the paper

We introduce the following notation for a function $\phi$ and accuracy $\varepsilon \geq 0$:

$$
\begin{aligned}
z \approx_\varepsilon \arg\min_z \phi(z) \quad &\Leftrightarrow \quad \phi(z) \leq \min_z \phi(z) + \varepsilon \\
z \approx_\varepsilon \arg\max_z \phi(z) \quad &\Leftrightarrow \quad \phi(z) \geq \max_z \phi(z) - \varepsilon \\
z \approx_\varepsilon \mathtt{prox}_{\tau\phi}(\bar{z}) \quad &\Leftrightarrow \quad z \approx_\varepsilon \arg\min_z \phi(z) + \frac{1}{2\tau}||z - \bar{z}||^2
\end{aligned}
$$

The rest of the paper is organized as follows. The next section describes Frank-Wolfe algorithms for minimizing a smooth function over a convex polytope. Then in Section 3 we will present our first approach, which is based on an inexact accelerated proximal point algorithm on the dual problem. In Section 4 we present our second approach, which is based on an inexact proximal primal-dual algorithm and directly solves the saddle point problem. Preliminary numerical results are given in Section 5. More technical proofs can be found in the suppl. material.

## 2. Frank-Wolfe algorithms

Frank-Wolfe style algorithms is a class of algorithms for minimizing functions $g_\mathcal{P} : \mathcal{X} \to \mathbb{R}$ of the form $g_\mathcal{P}(x) = g(x) + \delta_\mathcal{P}(x)$ where $g$ a convex continuously differentiable function with a Lipschitz continuous gradient and $\mathcal{P}$ is a convex polytope. They are typically iterative techniques that work by applying a certain procedure $\mathtt{FWstep}(x; g_\mathcal{P}) \mapsto x'$ where $g_\mathcal{P}$ is the objective function, and $x$ and $x'$ are respectively the old and the new iterates with $g_\mathcal{P}(x') \leq g_\mathcal{P}(x)$. We will apply such steps to functions $g_\mathcal{P}$ that change from time to time, which is why $g_\mathcal{P}$ is made a part of the notation.

It will be convenient to denote $g^\downarrow_\mathcal{P}(x) = g_\mathcal{P}(x) - \min_{x \in \mathcal{X}} g_\mathcal{P}(x)$ to be a shifted version of $g_\mathcal{P}$ with $\min_{x \in \mathcal{X}} g^\downarrow_\mathcal{P}(x) = 0$. The following fact is known.

**Lemma 1** (Lacoste-Julien & Jaggi (2015))**.** *For a point*

$\hat{x} \in \mathcal{P}$ *denote* $\texttt{gap} = \texttt{gap}^{\texttt{FW}}(\hat{x}; g_{\mathcal{P}}) = \langle \nabla g(\hat{x}), \hat{x} - s \rangle$ *where* $s = \texttt{lmo}_{\mathcal{P}}(\nabla g(\hat{x}))$. *Then*

$$g^{\downarrow}_{\mathcal{P}}(\hat{x}) \le \texttt{gap} \le \begin{cases} g^{\downarrow}_{\mathcal{P}}(\hat{x}) + LD^2/2 & \textit{if } g^{\downarrow}_{\mathcal{P}}(\hat{x}) > LD^2/2 \\ D\sqrt{2L \cdot g^{\downarrow}_{\mathcal{P}}(\hat{x})} & \textit{if } g^{\downarrow}_{\mathcal{P}}(\hat{x}) \le LD^2/2 \end{cases}$$

*where $D$ is the diameter of $\mathcal{P}$ and $L = L_g$ is the Lipschitz constant of $\nabla g$.*

While the original FW algorithm has a sublinear convergence rate, there are several variants that achieve a linear convergence rate under some assumptions on $g$. Examples include *Frank-Wolfe with away steps* (AFW) (Lacoste-Julien & Jaggi, 2015), *Decomposition-invariant Conditional Gradient* (DiCG) (Garber & Meshi, 2016; Bashiri & Zhang, 2017), and *Blended Conditional Gradient* (BCG) (Braun et al., 2019). Each step in these methods is classified as either *good* or *bad*. Good steps are guaranteed to decrease $g^{\downarrow}_{\mathcal{P}}(x)$ by a constant factor. Bad steps do not have such guarantee (because they hit the boundary of the polytope), but they make $x$ "sparser" in a certain sense and thus cannot happen too often.

More formally, consider a class of functions $\mathfrak{F}$ where each function $g_{\mathcal{P}} \in \mathfrak{F}$ is associated with a parameter vector $\Theta_{g_{\mathcal{P}}} \in \mathbb{R}^p$, and $\mathcal{P} = \operatorname{dom} g_{\mathcal{P}}$ is the same for all $g_{\mathcal{P}} \in \mathfrak{F}$. We say that procedure $\texttt{FWstep}$ has a *linear convergence rate on* $\mathfrak{F}$ if there exist continuous function $\theta : \mathbb{R}^p \to (0,1)$ and integers $R_0, R_1 \ge 0$ with the following properties: (i) if the step $\texttt{FWstep}(x; g_{\mathcal{P}}) \mapsto x'$ for $g_{\mathcal{P}} \in \mathfrak{F}$ is good then $g^{\downarrow}_{\mathcal{P}}(x') \le \theta(\Theta_{g_{\mathcal{P}}}) \cdot g^{\downarrow}_{\mathcal{P}}(x)$; (ii) when applying $\texttt{FWstep}(x; g_{\mathcal{P}})$ iteratively to some initial vector $x_0$ (possibly for different functions $g_{\mathcal{P}} \in \mathfrak{F}$), at any point we have $N_{\texttt{bad}} \le R_0 + R_1 N_{\texttt{good}}$ where $N_{\texttt{good}}$ and $N_{\texttt{bad}}$ are respectively the numbers of good and bad steps.

We will consider two classes of functions:

- $\mathfrak{F}_{\texttt{strong}} = \{ g_{\mathcal{P}}(x) = g(x) + \delta_{\mathcal{P}}(x)\ :\ g$ is a strongly convex differentiable function with a Lipschitz-continuous gradient, with $\Theta_{g_{\mathcal{P}}} = (\mu_g, L_g)\ \}$.
- $\mathfrak{F}_{\texttt{weak}} = \{ g_{\mathcal{P}}(x) = g(Ex) + \langle b, x \rangle + \delta_{\mathcal{P}}(x)\ :\ g$ is a strongly convex differentiable function with a Lipschitz-continuous gradient, and $E, b$ are matrix and vector of appropriate dimensions, with $\Theta_{g_{\mathcal{P}}} = (\mu_g, L_g, E, b)\ \}$.

Note that class $\mathfrak{F}_{\texttt{strong}}$ is implicitly parameterized by the dimension of vector $x$, and class $\mathfrak{F}_{\texttt{weak}}$ is implicitly parameterized by the dimensions of vector $x$ and matrix $E$.

The AFW method is known to have linear convergence on $\mathfrak{F}_{\texttt{strong}}$ (Lacoste-Julien & Jaggi, 2015) and also on $\mathfrak{F}_{\texttt{weak}}$ (Beck & Shtern, 2017; Lacoste-Julien & Jaggi, 2015). From the result of (Beck & Shtern, 2017; Lacoste-Julien & Jaggi, 2015) it is easy to deduce that the DiCG method with away steps also has linear convergence on $\mathfrak{F}_{\texttt{weak}}$, using, using Property 1 in (Bashiri & Zhang, 2017). The BCG method (Braun et al., 2019) has been shown to have linear convergence on class $\mathfrak{F}_{\texttt{strong}}$.

**Remark 1.** *Some of the techniques above maintain some additional information about current iterate $x$. In particular, AFW and BCG represent $x$ as a convex combination of "atoms" (vertices of $\mathcal{P}$): $x = \sum_i \alpha_i a_i$ where $\alpha_i \ge 0$, $\sum_i \alpha_i = 1$ and $a_i$ are atoms. Coefficients $\alpha_i$ are updated together with $x$. For brevity, we omitted this from the notation.*

**Remark 2.** *The claims about the number of bad steps are proven in (Lacoste-Julien & Jaggi, 2015; Garber & Meshi, 2016; Bashiri & Zhang, 2017; Braun et al., 2019) assuming that the function $g_{\mathcal{P}}$ is fixed. However, the proofs only use "structural" properties of current iterate $x$; they are easily extended to the case when $g_{\mathcal{P}}$ is changing, as long as $\mathcal{P}$ is fixed.*

**Iterative application of FWstep** Procedure $\texttt{FWstep}$ can be used in a natural way to solve problems $x \approx_\varepsilon \arg\min_x g_{\mathcal{P}}(x)$ up to desired accuracy $\varepsilon$.

**Algorithm 1** Algorithm $\texttt{FW}_\varepsilon(x; g_{\mathcal{P}})$.
**Output:** vector $x' \approx_\varepsilon \arg\min_x g_{\mathcal{P}}(x)$.

```
while true do
  update $x \leftarrow \texttt{FWstep}(x; g_{\mathcal{P}})$
  if $\texttt{gap}^{\texttt{FW}}(x; g_{\mathcal{P}}) \le \varepsilon$ then return $x$
end while
```

**Proposition 2.** *Suppose that procedure* $\texttt{FWstep}$ *has a linear convergence rate on class $\mathfrak{F}$ that contains $g_{\mathcal{P}}$. Then,*
*(a) The number of good steps made during* $\texttt{FW}_\varepsilon(x_0; g_{\mathcal{P}})$ *satisfies*

$$N_{\texttt{good}} \le \log_{1/\theta(\Theta_{g_{\mathcal{P}}})} \frac{g^{\downarrow}_{\mathcal{P}}(x_0)}{\min\{\frac{1}{2}LD^2, \frac{1}{2L}\left(\frac{\varepsilon}{D}\right)^2\}} \tag{4}$$

*where $D$ is the diameter of $\mathcal{P}$ and $L > 0$ is any constant satisfying $L \ge L_g$.*
*(b) Suppose that $g_{\mathcal{P}} \in \tilde{\mathfrak{F}} \subseteq \mathfrak{F}$ where $\sup_{g_{\mathcal{P}} \in \tilde{\mathfrak{F}}, x \in \mathcal{P}} g^{\downarrow}_{\mathcal{P}}(x) < \infty$, $\sup_{g_{\mathcal{P}} \in \tilde{\mathfrak{F}}} L_g < \infty$, and $\{\Theta_{g_{\mathcal{P}}} \mid g_{\mathcal{P}} \in \tilde{\mathfrak{F}}\}$ is a compact subset of $\mathbb{R}^p$. Then $N_{\texttt{good}} = O(\log \frac{1}{\varepsilon})$ where the constant in the $O(\cdot)$ notation depends on $\tilde{\mathfrak{F}}$.*

*Proof.* **(a)** By the definition of linear convergence, after the given number of good steps we obtain vector $x$ satisfying $g^{\downarrow}_{\mathcal{P}}(x) \le \min\{\frac{1}{2}LD^2, \frac{1}{2L}\left(\frac{\varepsilon}{D}\right)^2\}$. By Lemma 1, such $x$ satisfies $\texttt{gap}^{\texttt{FW}}(x; g_{\mathcal{P}}) \le \varepsilon$, and therefore the algorithm will immediately terminate.

**(b)** Since set $\{\Theta_{g_{\mathcal{P}}} | g_{\mathcal{P}} \in \tilde{\mathfrak{F}}\} \subseteq \mathbb{R}^p$ is compact and function $\theta : \mathbb{R}^p \to (0,1)$ is continuous, there exists $\theta^* \in (0,1)$ such that $\theta(\Theta_{g_{\mathcal{P}}}) \leq \theta^*$ for all $g_{\mathcal{P}} \in \tilde{\mathfrak{F}}$. Thus, all quantities present in (4) (except for $\varepsilon$) are bounded by constants for all $g_{\mathcal{P}} \in \tilde{\mathfrak{F}}$. The claim follows. □

## 3. First approach: dual proximal point alg.

The first approach that we consider is a proximal point algorithm (PPA) applied to the dual problem:

$$\max_{y \in \mathcal{Y}} \left\{ H(y) := \min_{x \in \mathcal{X}} \mathcal{L}(x, y) \right\}.$$

For a point $\bar{y} \in \mathcal{Y}$ and a smoothing parameter $\gamma > 0$, we let

$$\mathcal{L}_{\gamma,\bar{y}}(x, y) = \mathcal{L}(x, y) - \frac{1}{2\gamma}\|y - \bar{y}\|^2,$$

which can be seen as the original saddle-point problem, but with an additional proximal regularization on the dual variable. In each iteration, the PPA solves a maximization problem of the form $\hat{y} = \arg\max_{y\in\mathcal{Y}} H_{\gamma,\bar{y}}(y)$ where

$$H_{\gamma,\bar{y}}(y) := \min_{x\in\mathcal{X}} \mathcal{L}_{\gamma,\bar{y}}(x, y) = H(y) - \frac{1}{2\gamma}\|y - \bar{y}\|^2$$

Based on our structure, it will be beneficial to first solve for $\hat{x}$ and then to solve for $\hat{y}$ via its proximal map, that is

$$\begin{aligned} \hat{x} &= \arg\min_{x\in\mathcal{X}} \left\{ F_{\gamma,\bar{y}}(x) := \max_{y\in\mathcal{Y}} \mathcal{L}_{\gamma,\bar{y}}(x, y) \right\} \\ \hat{y} &= \arg\max_{y\in\mathcal{Y}} \mathcal{L}_{\gamma,\bar{y}}(\hat{x}, y) = \texttt{prox}_{\gamma h^*}(\bar{y} + \gamma K\hat{x}) \end{aligned}$$

Note that also for the smoothed saddle-point problem, strong duality holds,

$$\min_{x\in\mathcal{X}} F_{\gamma,\bar{y}}(x) = \min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \mathcal{L}_{\gamma,\bar{y}}(x, y) = \max_{y\in\mathcal{Y}} H_{\gamma,\bar{y}}(y),$$

and hence each step of the PPA can be equivalently written as minimizing the primal-dual gap

$$(\hat{x}, \hat{y}) = \underset{(x,y)\in\mathcal{X}\times\mathcal{Y}}{\arg\min}\ F_{\gamma,\bar{y}}(x) - H_{\gamma,\bar{y}}(y) \tag{5}$$

It is a well-known fact that the basic proximal point algorithm can be accelerated to achieve a $O(1/n^2)$ convergence rate (Güler, 1992; Salzo & Villa, 2012), which follows from the fact that the PPA can be seen as a steepest descent on the Moreau envelope (see suppl. material), which has a Lipschitz continuous gradient and hence can be accelerated using the technique of Nesterov (Nesterov, 1983).

However, based on our general assumptions on the problem (1), we will not be able to solve the proximal subproblems (5) exactly but only up to a certain error $\varepsilon > 0$ that is

$$(\hat{x}, \hat{y}) \approx_\varepsilon \underset{(x,y)\in\mathcal{X}\times\mathcal{Y}}{\arg\min}\ F_{\gamma,\bar{y}}(x) - H_{\gamma,\bar{y}}(y),$$

which clearly implies that $\hat{x} \approx_\varepsilon \arg\min_{x\in\mathcal{X}} F_{\gamma,\bar{y}}(x)$ as well as $\hat{y} \approx_\varepsilon \arg\max_{y\in\mathcal{Y}} H_{\gamma,\bar{y}}(y)$. However, we can still apply the recently proposed inexact accelerated proximal gradient algorithm of Aujol & Dossal (2015), that can handle such approximation while still achieving an optimal $O(1/n^2)$ convergence rate on the dual objective. Note that the original method given in (Aujol & Dossal, 2015) only generates the dual sequence $\{y_n\}$ but in Algorithm 2 below we also keep the primal sequence $\{x_n\}$ which is needed to obtain a solution of the original saddle-point problem (1). Therefore, the algorithm below can also be seen as a generalization for solving saddlepoint problems.

**Algorithm 2** Approx. accelerated proximal gradient method

```
choose nonnegative sequences $\{t_n\}$, $\{\varepsilon_n\}$ so that $t_1 = 1$
and $\rho_n \overset{\text{def}}{=} t_{n-1}^2 - t_n^2 + t_n > 0$ for all $n \geq 2$
choose initial point $y_0 \in \mathcal{Y}$, set $\bar{y}_0 = y_0$
for $n = 1, 2, \ldots$ do
    $(x_n, y_n) \approx_{\varepsilon_n} \underset{(x,y)\in\mathcal{X}\times\mathcal{Y}}{\arg\min}\ F_{\gamma,\bar{y}_{n-1}}(x) - H_{\gamma,\bar{y}_{n-1}}(y)$   (6)
    $\bar{y}_n = y_n + \frac{t_n - 1}{t_{n+1}}(y_n - y_{n-1})$   (7)
end for
```

In order to analyze this algorithm, let us introduce the following quantities:

$$\begin{aligned} &u_0 = y_0, \qquad u_n = y_{n-1} + t_n(y_n - y_{n-1}) \qquad \forall n \geq 1 \\ &A_n = \sum_{k=1}^{n} t_k \sqrt{2\gamma\varepsilon_k} \\ &B_n = \sum_{k=1}^{n} \gamma t_k^2 \varepsilon_k \\ &W_n = t_n^2[H(y^\star) - H(y_n)] + \sum_{k=2}^{n} \rho_k [H(y^\star) - H(y_{k-1})] \\ &T_n = t_n^2 + \sum_{k=2}^{n} \rho_k = \sum_{k=1}^{n} t_k \end{aligned}$$

First, we recall the following result from (Aujol & Dossal, 2015).

**Theorem 3** (Aujol & Dossal (2015)). *For any $n \geq 1$ there holds*

$$W_n + \frac{1}{2\gamma}||u_n - y^\star||^2 \quad \leq \quad \frac{C_n^*}{2\gamma}$$

*where*

$$\begin{aligned} C_n^* &= ||y_0 - y^\star||^2 + 2A_n \left(||y_0 - y^\star|| + 2A_n + \sqrt{2B_n}\right) + 2B_n \\ &\leq \left(||y_0 - y^\star|| + 2A_n + \sqrt{2B_n}\right)^2 \end{aligned}$$

This theorem immediately implies the following results. (Note, some of the statements below are slightly modified versions of statements from (Aujol & Dossal, 2015), but follow exactly the same proofs).

**Corollary 4** (Aujol & Dossal (2015))**.** *Suppose that sequences $\{A_n\}$ and $\{B_n\}$ are bounded. Then*
*(a) $H(y^\star) - H(y_n) = O(1/t_n^2)$.*
*(b) $H(y^\star) - H(y_n^e) = O(1/T_n)$ where $y_n^e = (t_n^2 y_n + \sum_{k=2}^n \rho_k y_{k-1})/T_n$.*
*(c) If function $-H(y)$ is coercive then sequence $\{y_n\}$ is bounded, and $||y_n - y_{n-1}|| = O(1/t_n)$.*

Note that the rate of convergence in Corollary 4 depends on the choice of sequence $\{t_n\}$. These are some of the choices that have appeared in the literature:

- PPA: $t_n = 1$ for all $n \geq 1$. Then $T_n = \Theta(n)$.
- Nesterov (Nesterov, 1983; Beck & Teboulle, 2009): $t_{n+1} = (1+\sqrt{1+4t_n^2})/2$ for $n \geq 1$. Then $t_n = \Theta(n)$ and $T_n = \Theta(n^2)$.
- Aujol-Dossal (Aujol & Dossal, 2015): $t_n = \left(\frac{n+a-1}{a}\right)^d$ with $d \in (0,1]$ and $a > \max\{1,(2d)^{1/d}\}$. Then $t_n = \Theta(n^d)$ and $T_n = \Theta(n^{d+1})$.

When stating complexities, we will implicitly assume below that either the second case or the third case with $d = 1$ is used, meaning that $t_n = \Theta(n)$ and $T_n = \Theta(n^2)$.

We now generalize Theorem 3 to the situation in this section. This generalization is somewhat analogous to the generalization obtained by Tseng (2008) (for a different Nesterov-type algorithm and with a different proof).

**Theorem 5.** *Denote $x_n^e = \sum_{k=1}^n t_k x_k / T_n$. For any $y \in \mathcal{Y}$ and any $n \geq 1$ there holds*

$$T_n \left[\mathcal{L}(x_n^e, y) - H(y^\star)\right] + W_n + \frac{1}{2\gamma}||u_n - y||^2 \leq \frac{C_n(y)}{2\gamma}$$

*where $C_n(y) = ||y_0 - y||^2 +$*
*$2A_n \left(||y - y^\star|| + ||y_0 - y^\star|| + 2A_n + \sqrt{2B_n}\right) + 2B_n$.*

Note that setting $y = y^\star$ in Theorem 5 recovers Theorem 3, since in this case we have $\mathcal{L}(x_n^e, y^\star) \geq H(y^\star)$ and $C_n(y^\star) = C_n^*$.

**Corollary 6.** *Suppose that sequences $\{A_n\}, \{B_n\}$ are bounded and $\operatorname{dom} h^* \subseteq \mathcal{Y}$ is a compact set. Then $F(x_n^e) - F(x^\star) = O(1/T_n) = O(1/n^2)$.*

*Proof.* By the assumption of the corollary, quantity $C_n(y)$ is bounded for any $y \in \operatorname{dom} h^*$ and $n \geq 1$. We also have $F(x_n^e) = \max_{y \in \operatorname{dom} h^*} \mathcal{L}(x_n^e, y)$ and $F(x^\star) = H(y^\star)$. The claim now follows directly from Theorem 5. □

Next, we analyze the special case of problem (1) corresponding to constrained optimization problem $\min_{x \in \mathbb{R}^d} \{f_\mathcal{P}(x) \mid Ax = b\}$.

**Theorem 7.** *Suppose we are in the case of the saddle problem in eq.* (2)*. (a) There holds*

$$f_\mathcal{P}(x_n^e) - f_\mathcal{P}(x^\star) \leq \frac{C_n(0)}{2\gamma T_n}$$

$$||Ax_n^e - b|| \leq \sqrt{\frac{2\max\{f_\mathcal{P}(x^\star) - f_\mathcal{P}(x_n^e), 0\}}{\gamma T_n}} + \frac{\hat{C}_n}{\gamma T_n}$$

*where $\hat{C}_n = ||y_0|| + A_n +$*
*$\sqrt{||y_0||^2 + 2A_n\left(||y^\star|| + ||y_0 - y^\star|| + 2A_n + \sqrt{2B_n}\right) + 2B_n}$.*
*(b) There exists constant $\beta \geq 0$ such that for any $x \in \mathcal{P}$ we have $f_\mathcal{P}(x^\star) - f_\mathcal{P}(x) \leq \beta ||Ax - b||$.*
*(c) If sequences $\{A_n\}$ and $\{B_n\}$ are bounded then $f_\mathcal{P}(x_n^e) - f_\mathcal{P}(x^\star) = O(1/n^2)$ and $||Ax_n^e - b|| = O(1/n^2)$.*

Note that part (a) is derived directly from Theorem 5. In part (b) we crucially exploit the facts that $\mathcal{P}$ is a polytope, the feasible set $\{x \in \mathcal{P} : Ax - b = 0\}$ is non-empty, and function $f$ has a bounded gradient on $\mathcal{P}$. Part (c) is an easy consequence of parts (a) and (b).

### 3.1. Overall algorithm

In this section we fix $n$, and denote $\bar{y} = \bar{y}_{n-1}$ and $\varepsilon = \varepsilon_n$. In order to implement Algorithm 2 for solving the saddle-point problem (1), we need to specify how to solve subproblem (6) for vector $\bar{y}$ up to accuracy $\varepsilon$:

$$(x_n, y_n) \quad \approx_\varepsilon \quad \arg\min_{(x,y) \in \mathcal{X} \times \mathcal{Y}} F_{\gamma,\bar{y}}(x) - H_{\gamma,\bar{y}}(y) \qquad (8)$$

We will first compute $x_n \approx_\varepsilon \arg\min_{x \in \mathcal{X}} F_{\gamma,\bar{y}}(x)$ by invoking Algorithm 1 for function $F_{\gamma,\bar{y}}$, and then solve for $y_n$ via its proximal map:

$$\begin{aligned} x_n &= \mathtt{FW}_\varepsilon(x_{n-1}; F_{\gamma,\bar{y}}) \\ y_n &= \arg\max_{y \in \mathcal{Y}} \mathcal{L}_{\gamma,\bar{y}}(x_n, y) = \mathtt{prox}_{\gamma h^*}(\bar{y} + \gamma K x_n) \end{aligned}$$

(As we will see later, function $F_{\gamma,\bar{y}}$ has the form $F_{\gamma,\bar{y}}(x) = g(x) + \delta_\mathcal{P}(x)$ for some differentiable convex function $g$ with a $L_g$-Lipschitz continuous gradient, and so Algorithm 1 is indeed applicable). By construction, vector $x_n$ satisfies $\mathtt{gap}^{\mathtt{FW}}(x_n; F_{\gamma,\bar{y}}) \leq \varepsilon$. The following lemma thus implies that the pair $(x_n, y_n)$ indeed solves problem (8).

**Lemma 8.** *Suppose that $\hat{x} \in \mathcal{P}$ and $\hat{y} = \arg\max_y \mathcal{L}_{\gamma,\bar{y}}(\hat{x}, y)$. Then $H_{\gamma,\bar{y}}(\hat{y}) \geq \mathcal{L}_{\gamma,\bar{y}}(\hat{x}, \hat{y}) - \varepsilon = F_{\gamma,\bar{y}}(\hat{x}) - \varepsilon$ where $\varepsilon = \mathtt{gap}^{\mathtt{FW}}(\hat{x}; F_{\gamma,\bar{y}})$.*

Next, we derive an explicit expression for function $F_{\gamma,\bar{y}}$ (which is needed for implementing the call $x_n =$

$\texttt{FW}_\varepsilon(x_{n-1}; F_{\gamma,\bar{y}})$), and formulate sufficient conditions on $\mathcal{L}$ that will guarantee that $F_{\gamma,\bar{y}} \in \mathfrak{F}_{\texttt{weak}}$ (this would yield a good bound on the complexity of Algorithm 2).

Recall that the function $F_{\gamma,\bar{y}}(x)$ is given by

$$F_{\gamma,\bar{y}}(x) = \max_{y \in \mathcal{Y}} \langle Kx,\, y\rangle + f_{\mathcal{P}}(x) - h^*(y) - \frac{1}{2\gamma}\|y-\bar{y}\|^2.$$

We now show that it can be written as

$$F_{\gamma,\bar{y}}(x) = f_{\mathcal{P}}(x) + h_{\gamma,\bar{y}}(Kx)$$

with $h_{\gamma,\bar{y}}(Kx)$ a differentiable function with Lipschitz continuous gradient.

**Lemma 9.** *Let the function $h_{\gamma,\bar{y}}(Kx)$ be defined as*

$$h_{\gamma,\bar{y}}(Kx) = \max_{y \in \mathcal{Y}} \langle Kx,\, y\rangle - h^*(y) - \frac{1}{2\gamma}\|y-\bar{y}\|^2.$$

*We have the following two representations:*

$$\begin{aligned} h_{\gamma,\bar{y}}(Kx) &= \frac{\gamma}{2}\|Kx\|^2 + \langle Kx,\, \bar{y}\rangle - m_{h^*}^{\gamma}(\bar{y}+\gamma Kx), \\ &= m_h^{\gamma^{-1}}\left(\gamma^{-1}\bar{y}+Kx\right) - \frac{1}{2\gamma}\|\bar{y}\|^2. \end{aligned}$$

*where $m_{h^*}^{\gamma}$ is the Moreau envelope of $h^*$ with smoothing parameter $\gamma$ and $m_h^{\gamma^{-1}}$ is the Moreau envelope of $h$ with smoothing parameter $\gamma^{-1}$.*
*Moreover, the function $h_{\gamma,\bar{y}}(Kx)$ is convex, continuously differentiable in $x$ with a $\gamma L_K^2$-Lipschitz continuous gradient given by*

$$\begin{aligned} \nabla_x h_{\gamma,\bar{y}}(Kx) &= K^* \texttt{prox}_{\gamma h^*}(\bar{y}+\gamma Kx) \\ &= \gamma K^*\big(\gamma^{-1}\bar{y}+Kx-\texttt{prox}_{\gamma^{-1}h}\left(\gamma^{-1}\bar{y}+Kx\right)\big) \end{aligned}$$

In practical applications, we will mostly be interested in the situation where $h^*$ is a linear constraint.

**Lemma 10.** *Let $\mathcal{X} = \mathcal{Y} = \mathbb{R}^\ell$, and let $h^*(y) = \delta_S(y)$ be a linear constraint of the form $S = \{y \in \mathcal{Y} : Cy = d\}$ for some matrix $C$ with full row rank and vector $d$. Then, the function $h_{\gamma,\bar{y}}(Kx)$ is a quadratic function of the form $h_{\gamma,\bar{y}}(Kx) = \frac{\gamma}{2}\|Kx\|^2 + \langle Kx,\, \bar{y}\rangle - \frac{1}{2\gamma}\|C^*(CC^*)^{-1}(d - C(\bar{y}+\gamma Kx))\|^2$. Moreover, its gradient is a linear map given by $\nabla_x h_{\gamma,\bar{y}}(Kx) = K^*\left(\bar{y}+\gamma Kx + C^*(CC^*)^{-1}(d-C(\bar{y}+\gamma Kx))\right)$.*

We therefore obtain the following sufficient condition where the functions $F_{\gamma,\bar{y}}(x)$ for any $\bar{y} \in \mathcal{Y}$ and any $\gamma > 0$ fall into the class $\mathfrak{F}_{\texttt{weak}}$.

**Lemma 11.** *Let $f(x)$ be a quadratic function of the form $f(x) = \frac{1}{2}\langle Qx,\, x\rangle + \langle q,\, x\rangle$, with a symmetric positive semidefinite matrix $Q$ and vector $q$. Furthermore, let $h^*$ satisfy the condition of Lemma 10. Then $\{F_{\gamma,\bar{y}} \mid \bar{y} \in \mathcal{Y}\} \subseteq \mathfrak{F}_{\texttt{weak}}$.*

*Proof.* First note that both $f(x)$ and $h_{\gamma,\bar{y}}(Kx)$ are quadratic functions. By completing the squares and ignoring constant terms, it follows that $F_{\gamma,\bar{y}}(x)$ can be written as $F_{\gamma,\bar{y}}(x) = \frac{1}{2}\|Ex\|^2 + \langle b,\, x\rangle + \delta_{\mathcal{P}}(x)$ for some matrix $E$ and vector $b$, where matrix $E$ may depend on $\gamma$ but not on $\bar{y}$. □

It remains to specify how to set the sequence $\{\varepsilon_n\}$. We want sequences $\{A_n\}$ and $\{B_n\}$ to be bounded; this can be achieved by setting $\varepsilon_n = \Theta(n^{-4-\delta})$ for some $\delta > 0$. With these choices, we obtain the main result of this section:

**Theorem 12.** *Suppose that function $\mathcal{L}$ satisfies the precondition of Lemma 11, function $-H(y)$ is coercive, and procedure* `FWstep` *has a linear convergence rate on $\mathfrak{F}_{\texttt{weak}}$ (e.g. it is one step of AFW or DiCG). Then Algorithm 2 makes $O(n\log n)$ calls to* `FWstep` *during the first $n$ iterations, and obtains iterates $x_n^e$ and $y_n^e$ satisfying $H(y^\star) - H(y_n^e) = O(1/n^2)$. Furthermore, in the case of the problem in eq.* (2) *the iterates satisfy $f_{\mathcal{P}}(x_n^e) - f_{\mathcal{P}}(x^\star) = O(1/n^2)$ and $||Ax_n^e - b|| = O(1/n^2)$.*

*Proof.* By Lemma 11, all functions $F_{\gamma,\bar{y}_{n-1}}$ encountered during the algorithm belong to $\mathfrak{F}_{\texttt{weak}}$. Furthermore, vectors $\bar{y}_{n-1}$ for $n \geq 1$ belong to a compact set, since the sequence $\{y_n\}$ (and thus the sequence $\{\bar{y}_n\}$) is bounded by Corollary 4(c). By Proposition 2(b) the number of good FW steps during $n$-th iteration is $O(\log \frac{1}{\varepsilon_n}) = O(\log n)$, and during the first $n$ iterations is $\sum_{k=1}^n O(\log k) = O(n \log n)$. The number of bad FW steps is thus also $O(n\log n)$ by the definition of linear convergence and by the fact that the call $x_n = \texttt{FW}_{\varepsilon_n}(x_{n-1}; F_{\gamma,\bar{y}_{n-1}})$ is initialized with vector $x_{n-1}$. The remaining claims follow from Corollaries 4 and 6 and Theorem 7. □

## 4. Second approach: primal-dual proximal algorithm

In this section we consider solving (1) without the restriction that $h^*$ is the indicator function of a linear constraint. Therefore we make use of proximal primal-dual algorithms such as (Chambolle & Pock, 2011) which in each step of the algorithm need to compute proximal maps with respect to $f_{\mathcal{P}}$ and $h^*$. By our problem assumptions, the proximal map with respect to $h^*$ is tractable but the proximal map with respect to $f_{\mathcal{P}}$ requires to solve for any $\bar{x} \in \mathcal{X}$ and $\tau > 0$ an optimization problem of the form

$$\texttt{prox}_{\tau f_{\mathcal{P}}}(\bar{x}) = \arg\min_{x \in \mathcal{P}} f(x) + \frac{1}{2\tau}\|x-\bar{x}\|^2.$$

We note the obvious fact that each proximal subproblem is a $\tau^{-1}$-strongly convex function with $L_f + \tau^{-1}$-Lipschitz continuous gradient over a convex polytope. Hence, it falls into the class $\mathfrak{F}_{\texttt{strong}}$, on which Frank-Wolfe algorithms achieve a linear rate of convergence. Similar to the previous section, we will not be able to solve the subproblems exactly but up to a certain accuracy $\varepsilon > 0$. We therefore need to resort to the recently proposed inexact primal-dual algorithm by Rasch & Chambolle (2020), Section 3.1 which can handle such inaccuracy. The algorithm adapted to our situation is given below as Algorithm 3.

**Algorithm 3** Inexact primal-dual algorithm.

```
choose $\tau, \sigma > 0$ such that $\sigma\tau L_K^2 < 1$
choose initial points $x_0 \in \mathcal{X}$ and $y_0 \in \mathcal{Y}$, set $x_{-1} = x_0$
for $n = 0, 1, \dots$ do
    $y_{n+1} \;=\; \mathtt{prox}_{\sigma h^*}(y_n + \sigma K(2x_n - x_{n-1}))$    (9)
    $x_{n+1} \approx_{\varepsilon_{n+1}} \mathtt{prox}_{\tau f_{\mathcal{P}}}(x_n - \tau K^* y_{n+1})$    (10)
end for
```

The following result has been shown in (Rasch & Chambolle, 2020). [1]

**Theorem 13** (Rasch & Chambolle (2020))**.**
*(a) Define* $x_n^e = \frac{1}{n}\sum_{k=1}^n x_k$ *and* $y_n^e = \frac{1}{n}\sum_{k=1}^n y_k$*. Then for any* $n \geq 1$ *and* $(x, y) \in \mathcal{X} \times \mathcal{Y}$

$$\mathcal{L}(x_n^e, y) - \mathcal{L}(x, y_n^e)$$

$$\leq \frac{1}{n}\left(\frac{||x - x_0||^2}{2\tau} + \frac{||y - y_0||^2}{2\sigma} + \frac{\mathtt{diam}(\mathcal{P})}{\tau} A_n + \frac{1}{\tau} B_n\right)$$

*where*

$$A_n = \sum_{k=1}^n \sqrt{2\tau\varepsilon_k} \qquad\qquad B_n = \sum_{k=1}^n \tau\varepsilon_k \qquad (11)$$

*(b) If sequences* $\{A_n\}$ *and* $\{B_n\}$ *are bounded then there exists saddle point* $(x^\star, y^\star)$ *of problem* (1) *such that* $x_n \to x^\star$ *and* $y_n \to y^\star$*.*

To solve the subproblem in eq. (10), we call Alg. 1 via $x_{n+1} \leftarrow \mathtt{FW}_{\varepsilon_{n+1}}(x_n; g_n)$ where $g_n(x) = f_{\mathcal{P}}(x) + \frac{1}{2\tau}||x - (x_n - \tau K^* y_{n+1})||^2$. To make sequences $\{A_n\}$ and $\{B_n\}$ bounded, we can set $\varepsilon_n = \Theta(n^{-2-\delta})$ for some $\delta > 0$. With these choices, we obtain

**Theorem 14.** *Suppose that procedure* `FWstep` *has linear a linear convergence rate on* $\mathfrak{F}_{\mathtt{strong}}$ *(e.g. it is one step of AFW, DiCG or BCG). Then Algorithm 3 makes* $O(n \log n)$ *calls to* `FWstep` *during the first* $n$ *iterations, and obtains iterates* $x_n^e$ *and* $y_n^e$ *satisfying* $F(x_n^e) - F(x^\star) = O(1/n)$ *(if* $\operatorname{dom} h^*$ *is a compact set) and* $H(y^\star) - H(y_n^e) = O(1/n)$*.*

## 5. Numerical results

In this section, we show preliminary results for solving MRFs arising from computer vision. The goal is to solve the following discrete minimization problem:

$$\min_{\mathrm{X} \in D^{\mathcal{V}}} E(\mathrm{X}) := \sum_{i \in \mathcal{V}} \theta_i(\mathrm{X}_i) + \sum_{ij \in \mathcal{E}} \theta_{ij}(\mathrm{X}_i, \mathrm{X}_j)$$

where $(\mathcal{V}, \mathcal{E})$ is a 4-connected 2D grid graph, $D$ is a finite set of labels, and $\theta_i(\cdot), \theta_{ij}(\cdot, \cdot)$ are given unary and pairwise costs, respectively. We decompose the problem into horizontal and vertical chains, and convert it to the saddle point problem (1) as described in Section 1.1.

We compare two versions of Algorithm 2: accelerated proximal point algorithm (A-PPA) with the aggressive choice $t_n = (n + 1)/2$ (which corresponds to the Aujol-Dossal scheme with $d = 1, a = 2$), and the standard proximal point algorithm (PPA) with the $t_n = 1$. Their convergence rates after $n$ iterations are $O(1/n^2)$ and $O(1/n)$ respectively, assuming that sequences $\{A_n\}$ and $\{B_n\}$ are bounded. We invoke Algorithm 1 to minimize the functions $F_{\gamma,\bar{y}}$ up to accuracy $\varepsilon_n = \mathtt{gap}_0 \cdot n^{-\alpha}$ for a constant $\alpha > 0$, where $\mathtt{gap}_0$ denotes the initial gap of the function $F_{\gamma,\bar{y}}$. Additionally, in the case of PPA we tested the version where we use a constant number ("`fw-it`") of Frank-Wolfe steps for the proximal subproblem. We view the latter version as a baseline, since this was the method suggested in (Swoboda & Kolmogorov, 2019).

Procedure `FWstep` was implemented as follows. We explicitly maintain the current iterate $x$ as a convex combination of atoms. In the beginning of `FWstep` we first run a standard Frank-Wolfe step, and then re-optimize the objective over the current set of atoms. For that one needs to solve a low-dimensional strongly convex, quadratic subproblem over the unit simplex; we used the linearly converging accelerated proximal gradient method described in (Nesterov, 2004). The resulting method can be viewed as a version of the BCG method (Braun et al., 2019), which is known to be linearly convergent on $\mathfrak{F}_{\mathtt{strong}}$.

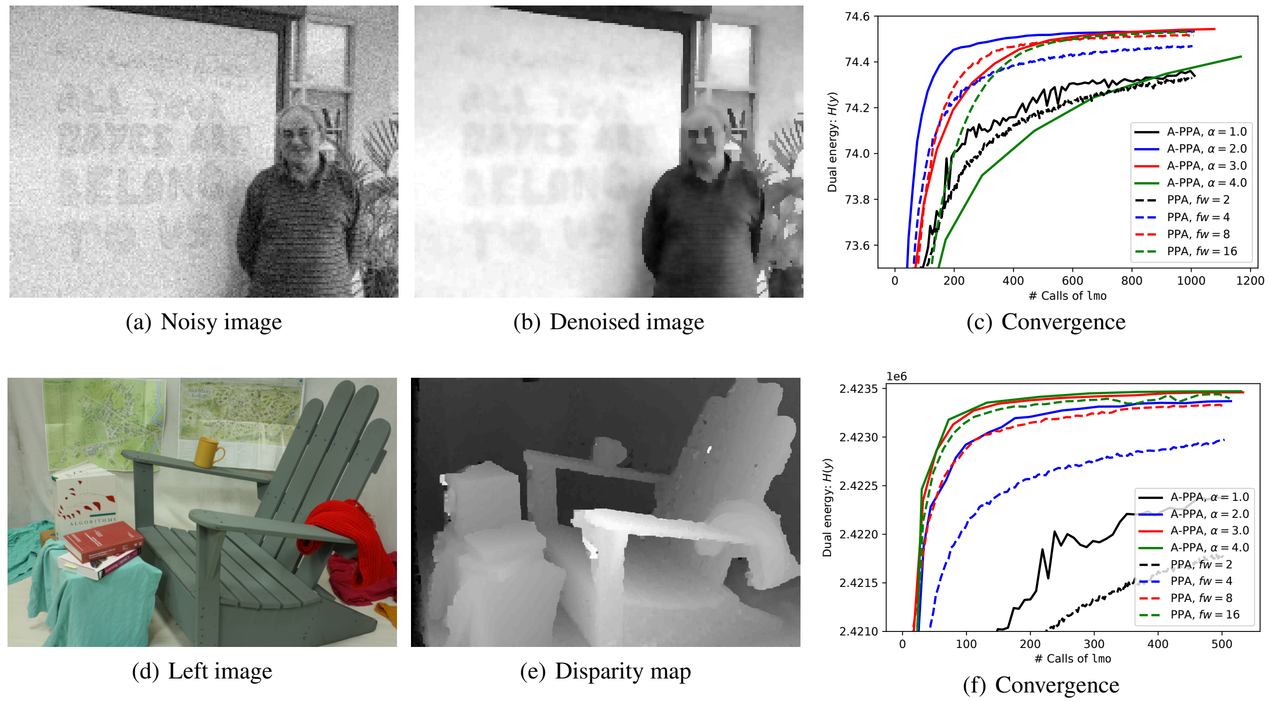
**Remark 3.** *Note that there is an extensive literature on FW variants. Potential alternatives to the method above include BCFW (Lacoste-Julien et al., 2013) and its variants (Shah et al., 2015; Osokin et al., 2016), DiCG (Garber & Meshi, 2016; Bashiri & Zhang, 2017), and Frank-Wolfe with in-face directions (Freund et al., 2017). However, testing different Frank-Wolfe methods is outside the scope of this paper; instead, we study what is the best way to use a given FW method.*

Next, we describe the two vision applications that we used.

**Image denoising** The first example is a simple image denoising problem. The unary terms are given by a quadratic potential function and the pairwise terms are given by a truncated quadratic potential function. Hence, this model resembles a discrete version of the celebrated Mumford-Shah functional. Figure 1 (a) shows the noisy input image which is of size $150 \times 200$ pixels and the image intensities are discretized using $50$ labels. Figure 1 (b) shows the denoised image.

**Stereo** The second example is a classical disparity estima-

[1] Part (a) is not formulated explicitly as a theorem in (Rasch & Chambolle, 2020), but can be found on page 396 before Theorem 2. Part (b) appears as Theorem 2 in (Rasch & Chambolle, 2020).



(a) Noisy image

(b) Denoised image

(c) Convergence

(d) Left image

(e) Disparity map

(f) Convergence

*Figure 1.* Results of the proposed algorithm for image denoising (first row) and disparity estimation (second row)

tion problem from a rectified stereo image pair (Scharstein et al., 2014). Figure 1 (d) shows the left input image which is of size $718 \times 496$ pixels. The disparities are discretized using 64 labels. The unary terms are pre-computed using a CNN-based correlation network (Knöbelreiter et al., 2020) and the pairwise costs are given by a truncated absolute potential function. The estimated disparity image is shown in Figure 1 (e).

**Results** Figure 1 (c) and (f) plot the convergence of the dual energy $H(y)$ over the total number of calls of lmo. The plots indicate that A-PPA clearly outperforms the baseline method PPA. Hence the theoretical improvement of this paper is also reflected in the practical performance. We also tested PPA with a prescribed accuracy $\varepsilon_n$ and it performed slightly better compared to a fixed number of iterations, but still worse compared to A-PPA. We refer to the supplemental material for such comparisons.

The results also indicate that the choice of $\alpha$ significantly influences the global convergence behavior. For example for $\alpha = 2$ A-PPA is fastest in the beginning but is catched up by $\alpha = 3$ after 500 calls of lmo. Note that in order to guarantee an $O(1/n^2)$ convergence rate for A-PPA we would have to set $\alpha > 4$, which seems to be competitive only for a very high accuracy. Therefore $\alpha$ should be chosen according to the desired accuracy of the solution. This is of particular interest if one is only interested in a fast approximate solution to the problem, for example if the MRF is used as the last inference layer in a CNN. For further experimental results we refer to the supplemental material.

## 6. Conclusion

In this work, we have proposed new primal-dual algorithms based on a mixture of proximal and Frank-Wolfe algorithms to solve a class of convex-concave saddle point problems arising in Lagrangian relaxations of discrete optimization problems. As our main result, we have shown after $O(n \log n)$ calls to lmo a $O(1/n)$ convergence rate in the most general case (Alg. 3) and a $O(1/n^2)$ convergence rate with certain regularity assumptions on the dual objective (Alg. 2). To the best of our knowledge, this improves on the known rates from the literature. Our preliminary numerical results also show an improved practical performance of Alg. 2 on MAP inference problems in computer vision. Note, we have not implemented yet Alg. 3 since its rate is worse on the application that we consider; at the moment the primary purpose of Alg. 3 is to show which rates are achievable for different classes of saddle point problems.

**Acknowledgements** Vladimir Kolmogorov was supported by the European Research Council under the European Unions Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement no 616160. Thomas Pock acknowledges support by an ERC grant HOMOVIS, no 640156.

## References

Argyriou, A., Signoretto, M., and Suykens, J. A. K. Hybrid conditional gradient-smoothing algorithms with applications to sparse and low rank regularization. In *Regularization, Optimization, Kernels, and Support Vector Machines*, pp. 53–82. Chapman and Hall/CRC, 2014.

Aujol, J.-F. and Dossal, C. Stability of over-relaxations for the forward-backward algorithm, application to FISTA. *SIAM Journal on Optimization*, 25(4):2408–2433, 2015.

Bashiri, M. A. and Zhang, X. Decomposition-invariant conditional gradient for general polytopes with line search. In *Conference on Neural Information Processing Systems (NIPS)*, 2017.

Beck, A. and Shtern, S. Linearly convergent away-step conditional gradient for non-strongly convex functions. *Mathematical Programming*, 164:1–27, 2017.

Beck, A. and Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.*, 2(1):183–202, 2009. ISSN 1936-4954.

Blake, A., Kohli, P., and Rother, C. *Markov random fields for vision and image processing*. MIT press, 2011.

Braun, G., Pokutta, S., Tu, D., and Wright, S. Blended conditional gradients: the unconditioning of conditional gradients. In *International Conference on Machine Learning (ICML)*, 2019.

Chambolle, A. and Pock, T. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vision*, 40(1):120–145, 2011. ISSN 0924-9907. doi: 10.1007/s10851-010-0251-1. URL `http://dx.doi.org/10.1007/s10851-010-0251-1`.

Condat, L., Kitahara, D., Contreras, A., and Hirabayashi, A. Proximal splitting algorithms: A tour of recent advances, with new twists. *arXiv preprint arXiv:1912.00137*, 2019.

Freund, R. M., Grigas, P., and Mazumder, R. An extended Frank-Wolfe method with "in-face" directions, and its application to low-rank matrix completion. *SIAM J. Optimization*, 27(1):319–346, 2017.

Garber, D. and Meshi, O. Linear-memory and decomposition-invariant linearly convergent conditional gradient algorithm for structured polytopes. In *Conference on Neural Information Processing Systems (NIPS)*, 2016.

Gidel, G., Jebara, T., and Lacoste-Julien, S. Frank-Wolfe algorithms for saddle point problems. In *Artificial Intelligence and Statistics*, pp. 362–371. PMLR, 2017.

Gidel, G., Pedregosa, F., and Lacoste-Julien, S. Frank-Wolfe splitting via augmented lagrangian method. In *AISTATS*, 2018.

Güler, O. New proximal point algorithms for convex minimization. *SIAM J. Optim.*, 2(4):649–664, 1992. ISSN 1052-6234. doi: 10.1137/0802032. URL `http://dx.doi.org/10.1137/0802032`.

Johnson, J., Malioutov, D. M., and Willsky, A. S. Lagrangian relaxation for map estimation in graphical models. In *45th Annual Allerton Conference on Communication, Control and Computing*, 2007.

Jojic, V., Gould, S., and Koller, D. Accelerated dual decomposition for MAP inference. In *International Conference on Machine Learning (ICML)*, 2010.

Knöbelreiter, P., Sormann, C., Shekhovtsov, A., Fraundorfer, F., and Pock, T. Belief propagation reloaded: Learning bp-layers for labeling problems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7900–7909, 2020.

Komodakis, N., Paragios, N., and Tziritas, G. MRF energy minimization and beyond via dual decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):531–552, March 2011.

Lacoste-Julien, S. and Jaggi, M. On the global linear convergence of Frank-Wolfe optimization variants. In *Conference on Neural Information Processing Systems (NIPS)*, 2015.

Lacoste-Julien, S., Jaggi, M., Schmidt, M., and Pletscher, P. Block-coordinate Frank-Wolfe optimization for structural SVMs. In *International Conference on Machine Learning (ICML)*, 2013.

Lan, G. and Zhou, Y. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2):1379–1409, 2016.

Liu, Y.-F., Liu, X., and Ma, S. On the non-ergodic convergence rate of an inexact augmented lagrangian framework for composite convex programming. *Mathematics of Operations Research*, 44:2, 2019.

Luong, D. V. N., Parpas, P., Rueckert, D., and Rustem, B. Solving MRF minimization by mirror descent. In *Advances in Visual Computing - 8th International Symposium, ISVC 2012, Rethymnon, Crete, Greece, July 16-18, 2012, Revised Selected Papers, Part I*, pp. 587–598, 2012.

Martins, A. F. T., Figueiredo, M. A. T., Aguiar, P. M. Q., Smith, N. A., and Xing, E. P. An augmented Lagrangian approach to constrained MAP inference. In *International Conference on Machine Learning (ICML)*, 2011.

Nesterov, Y. A method for solving the convex programming problem with convergence rate $O(1/k^2)$. *Dokl. Akad. Nauk SSSR*, 269(3):543–547, 1983. ISSN 0002-3264.

Nesterov, Y. *Introductory lectures on convex optimization*, volume 87 of *Applied Optimization*. Kluwer Academic Publishers, Boston, MA, 2004. ISBN 1-4020-7553-7. A basic course.

Osokin, A., Alayrac, J.-B., Lukasewitz, I., Dokania, P. K., and Lacoste-Julien, S. Minding the gaps for block Frank-Wolfe optimization of structured SVMs. In *International Conference on Machine Learning (ICML)*, 2016.

Rasch, J. and Chambolle, A. Inexact first-order primal–dual algorithms. *Computational Optimization and Applications*, 76:381–430, 2020.

Ravikumar, P., Agarwal, A., and Wainwright, M. J. Message-passing for graph-structured linear programs: Proximal methods and rounding schemes. *Journal of Machine Learning Research*, 11:1043–1080, 2010. doi: 10.1145/1756006.1756040. URL `http://doi.acm.org/10.1145/1756006.1756040`.

Salzo, S. and Villa, S. Inexact and accelerated proximal point algorithms. *J. Convex Anal.*, 19(4):1167–1192, 2012. ISSN 0944-6532.

Savchynskyy, B. Discrete graphical models – an optimization perspective. *Foundations and Trends® in Computer Graphics and Vision*, 11(3-4):160–429, 2019.

Savchynskyy, B., Kappes, J., Schmidt, S., and Schnörr, C. A study of nesterov's scheme for lagrangian decomposition and map labeling. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1817–1823. IEEE, 2011.

Savchynskyy, B., Schmidt, S., Kappes, J. H., and Schnörr, C. Efficient MRF energy minimization via adaptive diminishing smoothing. In *Uncertainty in Artificial Intelligence (UAI)*, 2012.

Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nesic, N., Wang, X., and Westling, P. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Pattern Recognition Conference (GCPR)*, pp. 31–42, 2014.

Schlesinger, M. I. and Giginyak, V. V. Solution to structural recognition (MAX,+)-problems by their equivalent transformations. (2):3–18, 2007.

Schmidt, S., Savchynskyy, B., Kappes, J. H., and Schnörr, C. Evaluation of a first-order primal-dual algorithm for mrf energy minimization. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pp. 89–103. Springer, 2011.

Schwing, A., Hazan, T., Pollefeys, M., and Urtasun, R. Globally convergent parallel MAP LP relaxation solver using the Frank-Wolfe algorithm. In *International Conference on Machine Learning (ICML)*, 2014.

Schwing, A. G., Hazan, T., Pollefeys, M., and Urtasun, R. Globally convergent dual MAP LP relaxation solvers using Fenchel-Young margins. In *NIPS*, 2012.

Shah, N., Kolmogorov, V., and Lampert, C. H. A multi-plane block-coordinate Frank-Wolfe algorithm for training structural SVMs with a costly max-oracle. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

Silveti-Falls, A., Molinari, C., and Fadili, J. Generalized conditional gradient with augmented lagrangian for composite minimization. *SIAM Journal on Optimization*, 30 (4):2687–2725, 2020.

Storvik, G. and Dahl, G. Lagrangian-based methods for finding map. *IEEE Trans. on Image Processing*, 9(3): 469–479, march 2000.

Swoboda, P. and Kolmogorov, V. MAP inference via block-coordinate Frank-Wolfe algorithm. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Tseng, P. On accelerated proximal gradient methods for convex-concave optimization. Technical report, University of Washington, Seattle, 2008.

Yurtsever, A., Fercoq, O., Locatello, F., and Cevher, V. A conditional gradient framework for composite convex minimization with applications to semidefinite programming. In *ICML*, 2018.